



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/828,465	04/21/2004	Shai Fultheim	1019-1002	5322

82253 7590 05/20/2011  
D. Kligler I.P. Services LTD  
P.O. Box 25  
Zippori, 17910  
ISRAEL

EXAMINER
----------

SILVER, DAVID

ART UNIT	PAPER NUMBER
----------	--------------

2128

NOTIFICATION DATE	DELIVERY MODE
-------------------	---------------

05/20/2011

ELECTRONIC

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

daniel@dkpat.co.il  
info@dkpat.co.il



UNITED STATES PATENT AND TRADEMARK OFFICE

---

Commissioner for Patents  
United States Patent and Trademark Office  
P.O. Box 1450  
Alexandria, VA 22313-1450  
[www.uspto.gov](http://www.uspto.gov)

**BEFORE THE BOARD OF PATENT APPEALS  
AND INTERFERENCES**

Application Number: 10/828,465  
Filing Date: April 21, 2004  
Appellant(s): FULTHEIM ET AL.

\_\_\_\_\_  
Daniel Kligler (Reg. No. 41,120)  
For Appellant

**EXAMINER'S ANSWER**

This is in response to the appeal brief filed on the 4th day of March, 2011 appealing from the Office action mailed 16th day of July, 2010.

Art Unit: 2128

**(1) Real Party in Interest**

The examiner has no comment on the statement, or lack of statement, identifying by name the real party in interest in the brief.

**(2) Related Appeals and Interferences**

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

**(3) Status of Claims**

The following is a list of claims that are rejected and pending in the application:

Claims 1-45.

**(4) Status of Amendments After Final**

The examiner has no comment on the appellant's statement of the status of amendments after final rejection contained in the brief.

**(5) Summary of Claimed Subject Matter**

The examiner has no comment on the summary of claimed subject matter contained in the brief.

**(6) Grounds of Rejection to be Reviewed on Appeal**

The examiner has no comment on the appellant's statement of the grounds of rejection to be reviewed on appeal. Every ground of rejection set forth in the Office action from which the appeal is taken (as modified by any advisory actions) is being maintained by the examiner except for the grounds of rejection (if any) listed under the subheading "WITHDRAWN REJECTIONS." New grounds of rejection (if any) are provided under the subheading "NEW GROUNDS OF REJECTION."

**(7) Claims Appendix**

The examiner has no comment on the copy of the appealed claims contained in the Appendix to the appellant's brief.

Art Unit: 2128

**(8) Evidence Relied Upon**

US 5,829,041	Okamoto	10-1998
--------------	---------	---------

US 20040054517	Altman	03-2004
----------------	--------	---------

VMware, "VMware Workstation - User's Manual - Version 3.2", Version 3.2, 9/5/2002, 420 pages

**(9) Grounds of Rejection**

The following ground(s) of rejection are applicable to the appealed claims:

1. Claims 1-4, 9-16, 22-30, and 32-45 are rejected under 35 U.S.C. 103(a) as being unpatentable over

Okamoto (**US 5829041**), in view of VMware Workstation "User's Manual" Version 3.2, ("VMware").

Okamoto discloses: 1. (currently amended) A method for executing a software application in a plurality of computers having respective hardware resources said hardware resources comprising a respective memory and a respective I/O device (**Fig 3 with emphasis on Computer 1 and Computer 2, as well as the implied networking between the computers; col: 2 line: 8-15**),

wherein said computers include a first computer and a second computer that intercommunication over a network (**col: 2 line: 8-15**), comprising the steps of:

running at least a first virtual machine implementer and a second virtual machine implementer on said first computer and said second computer, using said respective memory, wherein said first and second virtual machine implementers run separately and independently of one another on said first and second computers, respectively (**col: 5 line: 66 to col: 6 line: 7; the OS functions as the virtual machine implementer**).

Okamoto however does not expressly disclose:

executing a virtual machine on said computers, wherein sharing said virtual machine is shared between said first virtual machine implementer and said second virtual machine implementer using said respective I/O device in each of said first computer and said second computer to intercommunicate between said first computer and said second computer, and a guest operating system runs over said shared virtual machine.

Art Unit: 2128

Although Okamoto discloses executing a program executing transparently on the distributed system **(col: 5 line: 4-14)**, and functionally said collection of computers would function and the individual VM implementers (Oss) would function as a virtual machine, in order to make the record abundantly clear, VMware is used to show that such a program as disclosed by Okamoto could be VMware's Workstation (Virtual Machine), and such program would be distributed among the computers in the fashion as disclosed by Okamoto, also that such program would execute a virtual machine with a guest operating system.

Specifically, VMware discloses executing a program which creates a virtual machine and contains a guest operating system. The combination of the two references would yield predictable results **(VMWare: page 21)**.

It would have been obvious to one of ordinary skill in the art at the time of Appellant's invention to combine the references in order to achieve the various goals of virtual computing, such as, for example, sandboxing and security afforded to virtual environments. Further motivations include crash-resistance and crash-recovery, which would benefit from the VM environment. In fact, Okamoto provides clear motivation showing that a single program (VMware's Virtual Machine, for example), creates "easier" management when many computers execute one program **(col: 5 line: 36-42)**.

The combination teaches: 2. (previously presented) The method according to claim 1, further comprising step of running said software application over said guest operating system, so that commands invoked by said software application are monitored or emulated by said first virtual machine implementer and said second virtual machine implementer on said first computer and said second computer, while said hardware resources of said first computer and said second computer are shared by communication over said network **(VMWare: page 21; this is inherent as the commands executed would be loaded into a single aggregated virtual space, and then properly executed with the respective computer and resource. This is evidenced by Okamoto: col: 5 line: 66 to col: 6 line: 7)**.

Art Unit: 2128

The combination teaches: 3. (original) The method according to claim i, wherein at least one of said first virtual machine implementer and said second virtual machine implementer is a virtual machine monitor

**(Okamoto: col: 5 line: 66 to col: 6 line: 7; this is the function of Okamoto's OS).**

The combination teaches: 4. (original) The method according to claim i, wherein at least one of said first virtual machine implementer and said second virtual machine implementer is an emulator **(Okamoto:**

**col: 5 line: 66 to col: 6 line: 7 Okamoto's OS emulates a memory space that does not exist).**

9. (previously presented) The method according to claim i, further comprising the steps of:

providing a management system for said first virtual machine implementer and said second virtual machine implementer to control said first computer and said second computer, respectively, wherein said management system comprises a wrapper for receiving calls to a device driver from said first virtual machine implementer, said wrapper invoking said device driver according to a requirement of said first virtual machine implementer **(col: 16 line: 52-62: the OS functions as the management and the VMI, which must use device drivers to communicate with the memory of the first and second computer, and consequently functions as a wrapper for calls made to the virtual memory space, and direct them to the actual device drivers and actual memory space).**

VMware discloses: 10. (previously presented) The method according to claim 9, further comprising the step of providing a virtual PCI controller for said management system to control a physical PCI controller in one of said computers **(VMware page 206; 247 - bottom).**

VMware discloses: 11. (previously presented) The method according to claim 9, further comprising the step of providing a virtual DMA controller for said management system to control a physical DMA controller in one of said computers **(VMware page 149).**

VMware discloses: 12. (previously presented) The method according to claim 11, further comprising the steps of:

providing a virtual PCI controller to control a physical PCI controller in one of said computers; and during a bootup phase of operation scanning a device list with said virtual PCI controller to identify devices having on-board DMA controllers **(VMware page page 149-150 top).**

Art Unit: 2128

Okamoto discloses: 13. (previously presented) The method according to claim 1, further comprising the steps of:

with said first virtual machine implementer and said second virtual machine implementer maintaining mirrors of a portion of said respective memory that is used by said guest operating system in each of said computers; write-invalidating at least a portion of a page of said respective memory in one of said computers; and transferring a valid copy of said portion of said page to said one computer from another of said computers via said network (**Okamoto: col: 11 line: 59 to col: 12 line: 10**).

As per claims 14-16, 22-30, 32-45, note the rejection of claims 1-4, 9-13 above. The Instant Claims recite substantially same limitations as the above-rejected claims and are therefore rejected under same prior-art teachings.

2. Claims 5-8, and 17-21, 31 are rejected under 35 U.S.C. 103(a) as being unpatentable over Okamoto (**US 5829041**), in view of VMware Workstation "User's Manual" Version 3.2, ("VMware") in view of Altman (**US 20040054517**).

As per claim 5, the combination of Okamoto and VMware fully disclose claim 1. The combination however does not expressly disclose: at least said first computer comprises a first virtual node comprising a first physical CPU of said first computer and a second virtual node comprising a second physical CPU of said first computer.

Altman however discloses said feature (**para 0046**).

It would have been obvious to one of ordinary skill in the art at the time of Appellant's invention to combine the references in order to support multiprocessor environments, which are typically faster and cost less as technology advances. Using multiple cores / multiple processors allows for more compact machines which allows for more space to be used for additional computers, or other necessary features (such as cooling fixtures).

Art Unit: 2128

As per claims 6-8, note the rejection of claim 5 above. The Instant Claims recite substantially same limitations as the above-rejected claim and are therefore rejected under same prior-art teachings.

As per claims 17-21, note the rejection of claims 5-8 above. The Instant Claims recite substantially same limitations as the above-rejected claims and are therefore rejected under same prior-art teachings.

As per claim 31, note the rejection of claim 5 above. The Instant Claim recites substantially same limitations as the above-rejected claim and is therefore rejected under same prior-art teachings.

### **(10) Response to Argument**

#### **10.1 Appellants argue:**

10.1.1 In section (7).1.A (pages 4-6 of the Appeal Brief), Appellants state that the Examiner did not review secondary consideration not refute the substance of secondary consideration.

#### **10.2 Examiner Response:**

10.2.1 This is factually incorrect. The Declarations introduced through the prosecution of the Instant Application were all considered timely but were found to not have sufficient weight for overcoming the applied rejection. Attention is respectfully drawn to, for example, pages 2-7 of the Final Office Action dated 7/16/2010. As this Action is part of the record, it shall not be reproduced to not belabor the point. However, provided is an exemplary summary for why the Declarations were not persuasive. The Declaration appears to dive into the minds of the authors of the VMware and make conclusionary statements such as "the very authors of the VMware User Manual [] do not regard their own product as a possible solution of the types of computing configurations described and claimed"; however, the author of the Declaration is not an author of the VMware reference. Therefore, making such conclusionary statements is unpersuasive. The Declaration is replete with instances of leaping to conclusion without factual evidence, as pointed-out in the Office Action dated 7/16/2010.

10.2.2 It is again stressed that all of the evidence that was submitted was considered. However, it does not outweigh the set-forth rejections.



10.3 **Appellants argue:**

10.3.1 In section (7).1.B (pages 6-8), Appellants state that the "Examiner has failed to meet the requirements of MPEP 716.01(B) with respect to the objective evidence of non-obviousness raised in the Declarations by Landman and Yehuda." Appellants go on to cite various related MPEP sections.

10.3.2 "The Examiner has not commented on (and has barely even acknowledged) the declarations by Dr. Landman and Mr. Yehuda, and certainly has not given an explanation to support his findings regarding the evidence of secondary considerations provided by these declarations." (Brief: page 7)

10.3.3 "Although the Examiner in the present application appears to have taken the position that he may consider or disregard Appellant's objective evidence of non-obviousness at his option, this position is in outright contradiction to the dictates of MPEP and the case law." (Brief: page 8)

10.4 **Examiner Response:**

10.4.1 Regarding subsection 1-2 *supra*, the evidence was indeed timely considered at the time of its submission. Nothing in the MPEP dictates the every point of the Declaration must be readdressed in the Office Action. Indeed, only consideration is required, which was given. It was not sufficient to overcome the weight of the rejection when considering the evidence in its totality.

10.4.2 Regarding the Landman declaration, the declaration was to show that "the references that the Examiner has cited against the claims in the Application could not have led a person of ordinary skill in the art to make the claimed invention". This point is moot since the rejections of the Instant Action are based on different art.

The Declaration was also to show that the claims answer "a market need that could not be satisfied by prior art solutions and has been enthusiastically received by market leaders and customers". There are absolutely no metrics, nor sales numbers by which to judge the second

Art Unit: 2128

attempted point of the Declaration, specifically, the addressing of a long felt-need. In point 17 there is a mention of an imitator, 3Leaf, however, there is no direct correlation of their product to the Instant Claims. Much less establishing a timeline for when such an imitator emerged. The Declaration is filled with subjective conclusionary statements such as (taken from point 15): "vSMP has overcome these limitations and is rapidly growing in use and recognition." However, there is no bar by which to set or compare its "rapid" growth. In fact, "rapid" growth may be extremely slow when compared to field of virtualization or distributed computing.

10.4.3 Reviewing the Yehuda declaration, it is apparent that although Mr. Yehuda was indeed involved "in the computer industry", his primary career revolved around sales, as demonstrated by his curriculum vita. In section 3) he mentions that he was in the computer industry for 23 years, 6 of which he worked at Sun Microsystems. The Declarations then states that during this time he had learned about ScaleMP. It is not clear when he learned about ScaleMP. If this was on year 23 of his career, then clearly multi-processor have been in existence, which would defeat the statements made in sections 4)-6). This is indeed possible, since Mr. Yehuda was involved in sales, not the actual computer engineering field; certainly not virtual computing, or distributed computing fields. Further, examining the Declaration as a whole (with emphasis on section 5), there is absolutely no mention of when the vSMP was first shown to Mr. Yehuda as "working." Lack of concrete dates of when the demonstration occurred, lack of actual correlation to what was shown (or even discussed) to the claimed invention, lack of factual evidence to the conclusionary statements of "highly attractive for end-users". There are no sales numbers, nor metrics to compare this highly subjective declaration's statements. It appears to be drawn to a Declaration showing a long-felt need, however, this cannot be deduced from therein.

10.4.4 The Declarations do not address to the most recent rejection and at best provide conclusionary statements without factual basis.

10.4.5 The Declarations were, when timely submitted, timely considered but found insufficient to overcome the current rejection.

10.5 **Appellants argue:**

10.5.1 "MPEP and the case law make clear that objective evidence of non-obviousness (secondary considerations) submitted by the applicant must be considered and given due weight by the Patent Office. In the present case, however, the Examiner has chosen not to give Appellant's evidence any weight at all.

10.5.2 Proper consideration of the evidence that Appellant has submitted in the present case leads to a clear conclusion that the claimed invention is non-obvious. There is no evidence on the record, nor even a reasoned explanation, that could be taken in any way to refute Appellant's evidence and arguments in this regard. Even putting aside the question of *prima facie* obviousness (which Appellant does not concede by any means), the rejection of the claims in this application should be overturned because they are objectively non-obvious, and the application should therefore be allowed." **(In section (7).1.C (page 8))**

10.6 **Examiner Response:**

10.6.1 Regarding subsection 1 *supra*, "due weight" does not mean "sufficient weight to overcome the prior-art". In this instance, the weight given to the evidence was insufficient to overcome the rejections. If no weight was given, which is not conceded, it was due to the vague language, and lack of factual support to support statements made in the Declaration. The Declarations make leaps of "rapid growth" but do not correlate the growth of the actual claimed invention, nor shows its growth relative to the entire market of virtualization / distributed computing.

10.6.2 Regarding subsection 2 *supra*, the statement is conclusionary at best for the reasons set-above.

10.7 **Appellants argue:**

10.7.1 "The key point, as explained at length in the attached declarations by Landman, Tel-Zur and Eylon, is that a virtual machine is shared between multiple computers. The virtual machine (shown as VM 20 in the above figure) is supported by separate and independent virtual machine

Art Unit: 2128

implementers on the different computers (shown as virtual machine monitors - VMMs 34, 36, 38 - on computers 22, 24, 26). A guest operating system (GUEST OS 18) runs over the shared virtual machine, and software applications (APPLICATION 12, 14, 16) execute on the guest operating system.

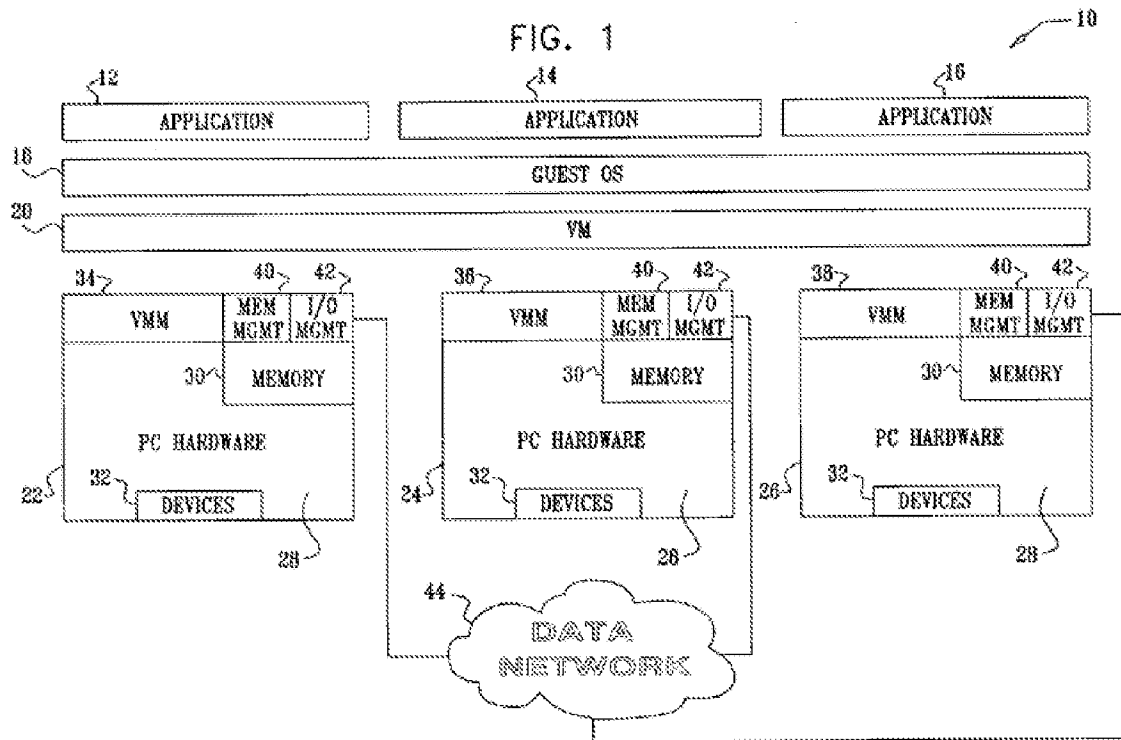
10.7.2 In other words, as explained by the declarants, the shared virtual machine appears to the guest operating system and the application software as if it was a single, high-powered computer, even though it uses the resources of multiple physical computers. The term "virtual machine" is a term of art, which is defined as follows in the VMware reference that was cited by the Examiner: A virtual machine "is equivalent to a PC with a unique network address and a full complement of hardware devices" (emphasis added)." (Brief: page 9 bottom to page 10)

10.7.3 "Comparing Okamoto's figures and description to claim 1 and to Fig. 1 shown on the preceding page, striking differences immediately become evident: Okamoto has no virtual machine at all - not in the sense in which the term is understood in the art or defined by the very VMware reference that the Examiner himself cited. In the absence of any sort of virtual machine, it is clear that Okamoto cannot and does not teach or suggest a virtual machine shared among his multiple physical computers, as illustrated in Fig. 1. Furthermore, in the absence of a virtual machine, Okamoto has no use for virtual machine implementers, nor can he support a guest operating system, both of which are shown in Fig. 1 and are required by claim 1. In the final analysis, the only commonality between Okamoto and the claimed invention is that both have the elements recited in the claim preamble: a plurality of computers connected by a network executing a software application, along with the unfortunate semantic coincidence of using the same term "virtual" to mean different things." (Brief: page 11; emphasis in original)

10.8 **Examiner Response:**

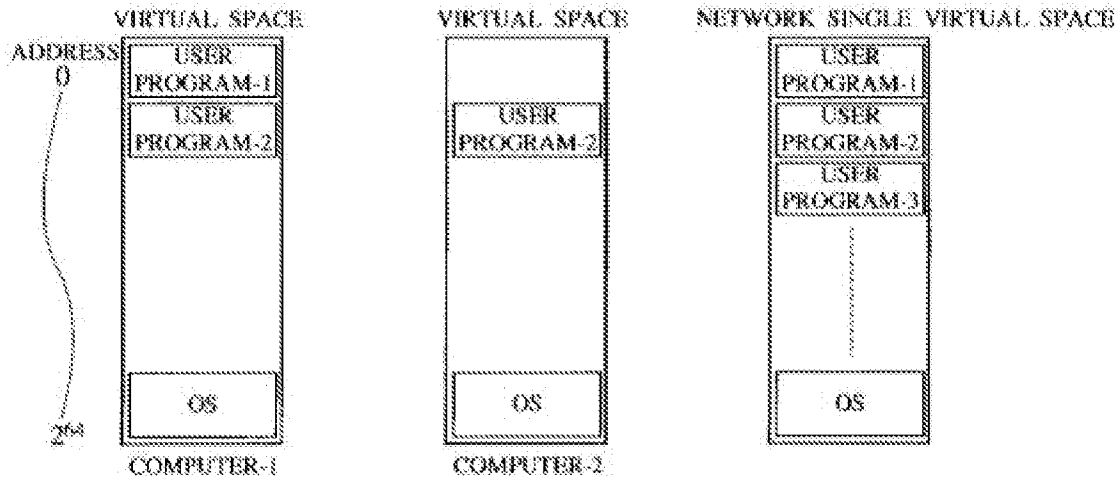
Art Unit: 2128

10.8.1 The claims, as demonstrated by Fig. 1 of the Instant Application, recite a multiple-computer system coupled via a network interface. See Fig. 1 reproduced below:



10.8.2 Regarding subsection 3 *supra*, absolutely no assertion was made in the Office Action that Okamoto has a virtual machine. Instead, Okamoto is much more powerful, as it allows for a distributed processing environment that can execute any program on multiple machines. This is indeed shown by Fig. 2. of Okamoto, reproduced below:

FIG. 2



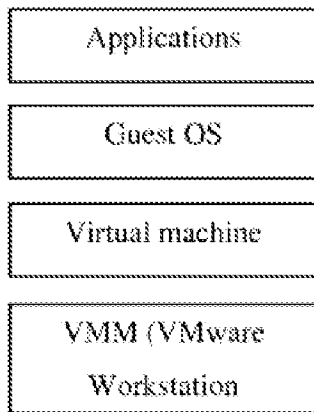
The "network single virtual space" comprises of the COMPUTER-1 and COMPUTER-2 executing, in this example, a "USER PROGRAM-2". This program is executed on both computer-1 and computer-2. This shows distributed computing. The user program can be any kind of application, whether a simple calculator, or a complex physics calculation, and does not need any kind of special programming (col: 5 line: 8-14: "application programs operable in the stand alone system can be operated under the distributed processing environment without any change in the application programs themselves, and consequently a programmer can produce the application programs without being conscious of a complicated virtual space management under the distributed processing environment").

10.8.3 Once again, it is noted that the user program 2 can be any application program running on the distributed processing environment (having multiple computers). The OS of Okamoto takes care of the management the resources.

10.8.4 It is the position of the Office Action that the User Program could have been VMware's Virtual Machine. Below is Appellants' rendition of VMware Workstation (taken from Brief, page 11 bottom). The implementer (VMware Workstation program) runs over a host operating system

Art Unit: 2128

and supports one or more virtual machines with guest operating systems. Indeed, VMware Workstation is not a piece of hardware, but software (a program). It can therefore, be executed in any computing environment, and given that it does not require one and only one computer, it can be executed on Okamoto's distributed computing environment. Given that it is a program, and using Appellants' own figures, it is shown as follows:

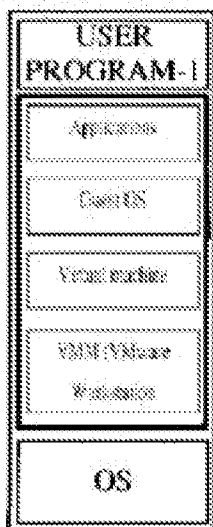


Where VMM is executed, which results in the execution of a Virtual Machine, which executes a Guest Operating System, which in turn executes various Applications.

Now, taking Fig. 2 of Okamoto, and substituting the above for VMware figure for USER

PROGRAM-2, the following is demonstrated:

#### NETWORK SINGLE VIRTUAL SPACE



Art Unit: 2128

Indeed, the Network Single Virtual Space is comprised of multiple computers running a Host Operating System (OS) (which function as a virtual machine implementers - there is no explicit definition of this term in the Specification), which executes a program (VMware Workstation), which creates a Virtual Machine, Guest OS, and runs multiple Applications. Now returning to the Fig 1 of the Instant Application. It appears clear that there are similarly, multiple computers, which have Virtual Machine Implementers (VMM, correlate to the OS of the Network Single Virtual Space of Okamoto), which run a single VM, a Guest OS, and multiple Applications. The only difference in the claimed invention, and the combination of the prior-art is that the combination allows for additional features, which the claims cannot handle. Specifically, the combination allows for more than the execution of a Virtual Machine, since Okamoto allows not only execution of a single User Program, but any number of user programs. The user program can similarly be any program, not just a virtual machine.

This is a simple substitution of a user program for another user program, which is a VMware's Workstation (which executes one or more virtual machines.)

#### 10.9 **Appellants argue:**

10.9.1.1 "This layered model is similar to that shown in Fig. 1 above, except that there is no hint in VMware that the virtual machines might possibly be shared among multiple virtual machine implementers or multiple computers. All of the expert declarations that Appellant has submitted make clear that despite the success and wide adoption of VMware products, it was not considered feasible to share virtual machines in this way prior to the present invention."  
(Brief: page 12)

10.9.2 "A person of ordinary skill in the art who was familiar with Okamoto and VMware might well have installed the VMware Workstation on each of Okamoto's separate computers, and then would have been able to run Okamoto's user programs over virtual machines and guest operating



Art Unit: 2128

systems on each computer individually. This is the result that the person of ordinary skill would have obtained.” (Brief: page 12; emphasis in original)

10.10 **Examiner Response:**

10.10.1 Indeed, VMware need not hint of multiple virtual machine implementers; otherwise, it would have been used in a 35 U.S.C. § 102 rejection, not a 35 U.S.C. § 103 rejection. Okamoto however, explicitly states that the program being executed need not be specially programmed to work on his distributed processing environment (**col: 5 line: 8-14: “application programs operable in the stand alone system can be operated under the distributed processing environment without any change in the application programs themselves, and consequently a programmer can produce the application programs without being conscious of a complicated virtual space management under the distributed processing environment”**).

10.10.2 Regarding subsection 2 *supra*, Appellants’ statement is completely opposite to Okamoto’s intent of having a distributed processing environment which can aggregate resources and leverage the power of multiple computers. It would make no sense to run a program on individual computers, as that would amount to bypassing the distributed processing environment all altogether. This is completely counter to what one of ordinary skill in the art would have obtained.

10.11 In view of KSR v. Teleflex Supreme Court ruling, it is asserted that the main reference differs from the claimed invention by a substitution of some components (User Program 2) with other components (VMware Workstation program), and that the substituted components were known (see VMware). It is further asserted that one of ordinary skill in the art could have substituted one known element User Program-2 for another VMWare Workstation program and the results of the substitution would have been predictable - it would have resulted in a distributed computer environment having a distributed virtual machine, sharing resources, among multiple computers. The virtual machine would execute one or more Guest Operating Systems, which in turn would execute their own applications.

Art Unit: 2128

10.12 There are many individual benefits and motivations for having a virtual machine, as well as a distributed environment, both of which leverage on their own advantages. For example, executing a virtual machine on a distributed environment allows it to leverage the power of multiple computers (this is a benefit of distributed computing environments.) One would have been additionally motivated to execute the VMware Workstation on the distributed computing environment because it leverages the power of multiple resources. Furthermore, it results in management becoming "easier even when many computers execute one program on the virtual spaces". **(col: Okamoto: 5 line: 40-42)**

See MPEP 2145 [R-6], X, B, "[A] person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely that product [was] not of innovation but of ordinary skill and common sense. In that instance the fact that a combination was obvious to try might show that it was obvious under § 103." KSR International Co. v. Teleflex Inc., 550 U.S. \_\_\_, \_\_\_, 82 USPQ2d 1385, 1397 (2007).

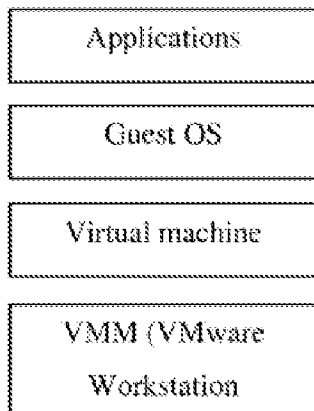
10.13 **Appellants argue:**

10.13.1 "The Examiner gave no specific grounds of rejection for claims 38 and 39, and stated merely the "The Instant Claims recite substantially same limitations as the above- rejected claims [1-4 and 9-13] and are therefore rejected under same prior-art teachings." In fact, however, none of claims 1-4 or 9-13 says anything at all about one instance of a single guest operating system. As noted earlier in reference to the independent claims, both Okamoto and VMware describe multiple operating system instances - at least one per computer." (Brief: page 13)

10.14 **Examiner Response:**

10.14.1 The issue of a Guest operating system and a Host operating systems is being confused by the arguments. There is only a single Guest Operating system being executed by the virtual machine. This is demonstrated by Appellants' rendition of VMware model, shown on Brief page 11.

Art Unit: 2128



There are however multiple Host operating systems, which function as the virtual machine implementers. Indeed, as demonstrated above and acquiesced by the Appellants, there is a single Guest operating system. There is not at least one Guest operating system per computer. Indeed, claims 1-4 and 9-13 recite substantially similar limitations.

10.15 **Examiner Summary:**

10.15.1 Appellants stated that Declarations were not considered. This is incorrect. All Declarations were considered.

10.15.2 Appellants stated that no weight was given to the Declarations. It was noted that the amount of weight given to the Declarations depended on the content of the Declarations. If a Declaration received little or no weight, it was solely based on the content and lack of factual evidence to support the subjective statements and leaps of conclusionary statements.

10.15.3 Appellants attempted to combine the references in ways that would clearly not be in the spirit of Okamoto and bypass the intent of Okamoto.

10.15.4 Arguments were presented that attempted to confuse the point of a Guest OS and Host OS.

**(11) Related Proceeding(s) Appendix**

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Art Unit: 2128

Respectfully submitted,

/ DAVID SILVER / 5/13/2011  
David Silver, Primary Patent Examiner, Art Unit 2128

Conferees:

/Kamini S Shah/

Supervisory Patent Examiner, Art Unit 2128

/Paul L Rodriguez/

Supervisory Patent Examiner, Art Unit 2123